Proceedings of the ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering
OMAE2015
May 31-June 5, 2015, St. John's, Newfoundland, Canada

OMAE2015-41282

THE HARMONIC POLYNOMIAL CELL METHOD FOR MOVING BODIES IMMERSED IN A CARTESIAN BACKGROUND GRID

Finn-Christian W. Hanssen
Centre for Autonomous Marine Operations and
Systems (AMOS), Department of Marine
Technology, Norwegian University of Science and
Technology, NTNU
NO-7491 Trondheim, Norway

Marilena Greco[1,2]
[1]Centre for Autonomous Marine Operations and
Systems (AMOS), Department of Marine
Technology, Norwegian University of Science and
Technology, NTNU
NO-7491 Trondheim, Norway
[2]CNR-INSEAN, Rome, Italy

Yanlin Shao
Ship Hydrodynamics & Stability,
DNV GL, Høvik, Norway

ABSTRACT

In numerical simulations with moving bodies, and often with complex geometries, generation of high-quality body-fitted grids is a cumbersome and time-consuming task. An alternative is to use a fixed (Cartesian) background grid, and allow the body to move freely over this. The challenge in such methods is to transfer the body-boundary conditions of the moving body to fixed grid nodes in a rational manner. In this paper, an Immersed Boundary Method (IBM) is proposed to simulate potential flow about a moving body on a Cartesian background grid. The recently developed Harmonic Polynomial Method, proven both accurate and computationally efficient, is used to represent the velocity potential in the fluid. The body-boundary conditions are interpolated by using ghost nodes inside the body with mirror interpolation points in the fluid.

The method is first tested for a fixed cylinder in oscillatory flow to determine the accuracy of the proposed IBM, before considering the equivalent case of an oscillating cylinder in still fluid. Finally, a steadily-advancing cylinder is studied, which is considered as the most challenging case with respect to spurious pressure oscillations. These are known to be a challenge in many IBMs, and special attention is therefore devoted to this aspect.

INTRODUCTION

In fluid-flow problems with significant nonlinear features, it is desirable to solve fully nonlinear hydrodynamic equations in time domain. For simulation of flow around submerged or semi-submerged bodies, the traditional approach is to use body-fitted grids that conform to the body boundary. For a moving body this means that, every time the body boundary moves, the grid needs to be updated accordingly. This may lead to difficulties in generating high-quality grids for challenging geometries and large boundary motions, and generally involves increased computational cost. In addition, assumptions that restrict the range of allowable motion may be necessary. As an alternative fixed-grid methods have been developed, where the grid nodes do not generally coincide with the body boundary. This approach offers significant gains in computational efficiency and eases the process of grid generation. The challenge in such methods is to obtain sufficient accuracy for the flow features at the body boundary.

For a fixed (Cartesian) grid, body-boundary conditions on immersed boundaries are projected onto surrounding nodes. Schemes for doing this are commonly denoted by Immersed Boundary Methods (IBM). Peskin [1] was the first to introduce this idea to simulate the viscous flow patterns around heart valves. The effect of the flexible immersed boundaries was replaced by a force field that only attained nonzero values in the immediate neighborhood of the immersed boundary. This approach represents a diffusive-interface (continuous forcing) method, where the boundary conditions effectively are smeared over a finite band of grid nodes. Later, IBMs belonging to the family of sharp-interface (discrete forcing) methods have emerged [2]. These generally offer higher accuracy than diffusive-interface methods, since the sharp nature of the boundary is better preserved.

A well-known challenge in sharp-interface IBMs is the occurrence of spurious oscillations in the fluid pressure, see e.g. [3], [4]. These are mainly related to the numerical evaluation of time derivatives. In continuous forcing methods, such as [1], oscillations are avoided more easily due to the diffuse nature of the interface. A sharp-interface method known as the cut-cell method may also be helpful in removing numerical oscillations. In this case cells that are intersected by boundaries are modified to conform to them. This method may however lead to accuracy issues when very small cells emerge from the cutting algorithm, which may be remedied by cell-merging schemes. However, such schemes may be nontrivial to implement, especially in three dimensions [5].

The present paper proposes a sharp-interface IBM for the recently established Harmonic Polynomial Cell (HPC) method for inviscid flows. This was developed by Shao & Faltinsen [6] - [8] as an efficient alternative to solve fully nonlinear potential flow problems. In [6], [7] it is demonstrated that the HPC method is highly competitive in terms of accuracy and computational efficiency compared to both a Boundary Element Method (BEM) and a Finite Volume Method (FVM). The method was further applied to sloshing in rectangular tanks, a fully nonlinear wave tank in two- and three dimensions and nonlinear wave diffraction around bottom-mounted cylinders. Fredriksen et al. [9] successfully coupled a HPC-based potential-flow solver with a laminar Navier-Stokes solver in two dimensions by applying a domain-decomposition strategy to study the gap resonance in a moonpool with small forward speed. Generally, their results were in good agreement with experiments. Liang et al. [10] applied the HPC method to study several additional problems relevant for marine hydrodynamics, including violent (nonlinear) sloshing in a two-dimensional tank. Very good agreement with experimental results was achieved indicating that, as long as the underlying assumptions of potential flow are valid, the method is capable of handling strongly nonlinear free-surface flows.

The accuracy, computational efficiency and relatively easy numerical implementation of the HPC method make it a well-suited tool to simulate flows around bodies undergoing arbitrary motions. Since an IBM preserves the ease of implementation as well as the computational efficiency of the HPC method, this seemingly represents a more attractive choice than a body-fitted grid for the purpose. Moreover, since the HPC method is based on a linear combination of polynomials, the additional interpolation required by an IBM is natural. The challenge is then to preserve the accuracy of the method in a numerical implementation.

In the present IBM, ghost nodes inside the body are applied to reconstruct the velocity and pressure at the immersed boundary. A concept is introduced to deal with time derivatives for nodes emerging from inside the body when the body is moving.

The paper is organized as follows: First, the numerical formulation is outlined, including a brief summary of the HPC method and a more thorough description of the deployed IBM. Then, the accuracy of the IBM is demonstrated for a two-dimensional cylinder in three conditions: fixed in an oscillatory inflow, oscillating and steadily advancing in still fluid. Finally, some concluding remarks and notions on further work are given.

NUMERICAL FORMULATION

Harmonic Polynomial Cell Method

A brief summary of the HPC method as formulated by Shao & Faltinsen is given. For an in-depth description, reference is made to [6], [7].

In the two-dimensional form of the HPC method, the computational domain is divided into overlapping quadrilateral cells. Each cell contains nine grid nodes: Node 1 – 8 along the boundaries and node 9 in the cell center. The velocity potential in each cell is written as a linear interpolation of the velocity potential φ_i in the eight nodes $i = 1,...,8$ along the cell's edges,

$$\varphi(x, y) = \sum_{i=1}^{8} \left[\sum_{j=1}^{8} c_{j,i} f_j(x, y) \right] \varphi_i \ . \tag{1}$$

Here $f_j(x, y)$ are the values of the eight harmonic polynomials listed in [6] evaluated in a generic (x, y) set coincident with the local cell coordinates; $c_{j,i}$ are elements of the inverse of the square matrix $[D]$ with elements $d_{i,j} = f_j(x_i, y_i)$. Each harmonic polynomial satisfies the Laplace equation, so that the velocity potential in each cell automatically satisfies the continuity equation for an inviscid, incompressible fluid:

$$\nabla \cdot \vec{u}(x, y) = \nabla^2 \varphi(x, y) = 0 \, . \tag{2}$$

The origin of the local coordinate system is chosen to coincide with the location of the ninth node in each cell. At the origin, $(x, y) = (0, 0)$, only $f_1 = 1 \neq 0$. Thus, for the central node in each cell the following equation applies:

$$\varphi_9 = \varphi(x = x_9 = 0, y = y_9 = 0) = \sum_{i=1}^{8} c_{1,i} \varphi_i \ . \tag{3}$$

Eq. (3) connects the entire computational domain, in that the velocity potential in every node is expressed as a combination of the potential in the eight surrounding nodes.

For nodes along the edges of the computational domain, either Dirichlet or Neumann conditions apply. A Dirichlet condition directly provides the value of the velocity potential, whereas a Neumann condition is given as

$$\frac{\partial \varphi}{\partial n}(x, y) = \sum_{i=1}^{8} \left[\sum_{j=1}^{8} c_{j,i} \nabla f_j(x, y) \cdot \vec{n}(x, y) \right] \varphi_i \ , \tag{4}$$

where $\vec{n}(x, y)$ is the two-dimensional normal vector in (x, y).

Immersed Boundary Treatment

For the flow around an impermeable body, a zero-flux Neumann condition applies at the boundary,

$$\nabla\varphi\left(x_b, y_b\right)\cdot\vec{n}\left(x_b, y_b\right)=\vec{U}\cdot\vec{n}\left(x_b, y_b\right), \qquad (5)$$

where (x_b, y_b) is a point along the body boundary with normal vector $\vec{n}(x_b, y_b)$ and $\vec{U}$ is the local rigid-body velocity.

Eq. (5) is straightforward to impose in a body-fitted grid. However, for a complex body geometry, it may be nontrivial to construct a suitable body-fitted grid in the first place. Moreover, for moving bodies, the grid must be updated at every time step. In the context of the HPC method, this implies that the elements of the local matrices in Eq. (1) have to be recomputed for each cell at every time step. Altogether, the process of re-gridding can significantly increase the computational time for dense grids or large computational domains. It should be noted that the computational burden due to re-gridding often can be reduced by only deforming the grid in a confined region close to the moving boundary. This does nonetheless put a restriction on the allowable motion amplitudes of the body.

The IBM described in the following belongs to the family of sharp-interface methods [2], using ghost nodes inside the immersed boundary to interpolate the body boundary conditions. The interpolation scheme is inspired by the ghost-cell method used by Tseng & Ferziger [11], although the fundamental flow equations differ. This section highlights key aspects of the IBM scheme.

Body Discretization

Uniformly distributed discrete Lagrangian marker points that move with the rigid-body velocity are used to track the boundary of the moving body. Linear segments between markers describe the body surface. The density of the background grid does not restrict the number of applied Lagrangian markers, but an excessive amount of markers may increase the computational time unnecessarily. Alternatively, a signed distance function such as used by Liu & Hu [12] could also have been used to track the position of the body boundary.

Node Identification

The first step in the applied IBM is to identify which grid nodes are inside or outside the body boundary. Further subdivision categorizes the nodes inside the body into four groups as illustrated by FIGURE 1: The first ghost-node layer contains the nodes that are next to the body boundary, the second layer contains the nodes inside the body that are next to the first layer and the third layer contains the nodes that are next to the second layer. The remaining interior nodes are without any specific purpose. However, we allow a fictitious interior flow to exist by enforcing Eq. (3) for these nodes, which appears to increase the solvability of the global matrix system somewhat.

For a moving body, node identification is required at every time step. This involves a double loop over all grid nodes and all Lagrangian body markers. A more efficient scheme is available by only considering grid nodes that are within a certain distance from the body's position at the previous time step, i.e. a confined search region. Liu & Hu [12] deployed this type of approach to limit the computational cost involved in tracking the body boundary in their level-set method.

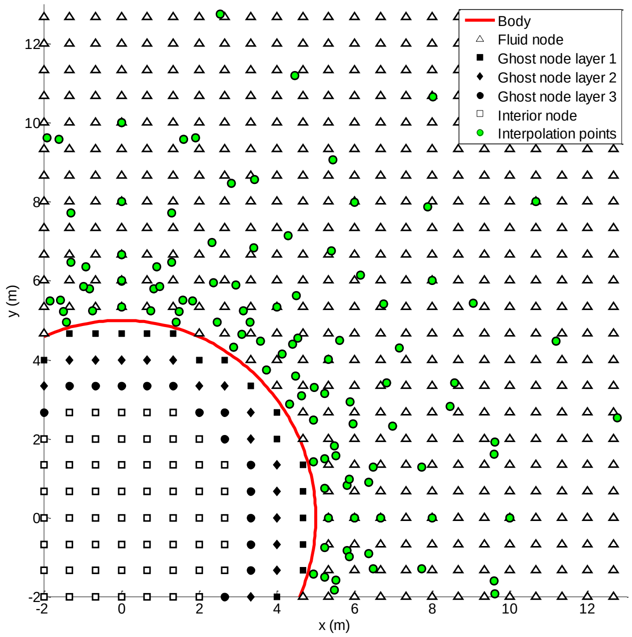


FIGURE 1: Ghost nodes and interpolation points for an example case with a medium dense background grid.

Body Boundary Condition

The body boundary condition in Eq. (5) is transferred to the three layers of ghost nodes by deploying an interpolation scheme along the direction of the body's local normal vector as illustrated in FIGURE 2. A cubic interpolation is chosen to interpolate the normal fluid velocity, giving third-order accuracy for the boundary condition in the spatial interpolation variable. The rationale behind this is to have an interpolation scheme of comparable order as the underlying HPC method, which has between third- and fourth order accuracy in the grid-separation distance [6].

The normal fluid velocity in a point (x, y) near the body boundary is approximated by a third-order polynomial,

$$\left.\nabla\varphi\right|_{(x,y)}\cdot\vec{n}\left(x_b, y_b\right)\approx a_0+a_1r+a_2r^2+a_3r^3. \qquad (6)$$

$\vec{n}(x_b, y_b)$ is the local normal vector related to a certain ghost node, and r is the distance from the ghost node along the direction of this normal vector. The normal vector for a given ghost node is taken as a weighted average of the normal vector of the body segment closest to the ghost node and one neighbor body segment on each side of this. This ensures a smooth variation of the body normal vectors as the body moves over the grid, which is found helpful in reducing numerical oscillations in the velocity potential.

The unknown coefficients in Eq. (6) are expressed through the normal velocity at the ghost node and at three uniformly distributed interpolation points in the fluid region (FIGURE 2). The distance between the points is $2s$, where s is the normal distance between the ghost node (x_g, y_g) and its normal projection (x_b, y_b) onto the body boundary. FIGURE 1 shows that, as a result, interpolation points are distributed at varying distances from the body boundary.

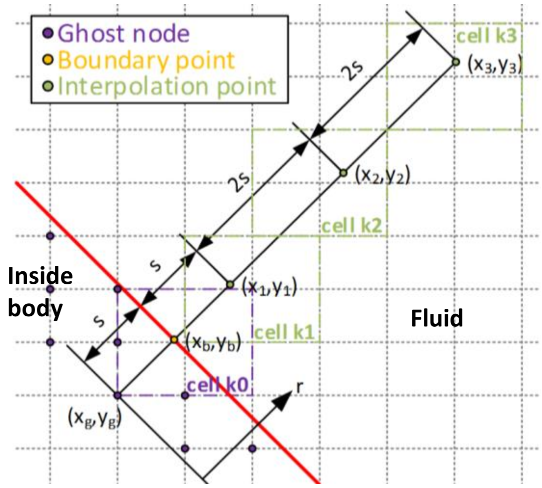


FIGURE 2: The boundary condition at the body point (x_b, y_b) is interpolated by using the ghost node (x_g, y_g) and three mirror points (x_{1-3}, y_{1-3}). The red line illustrates a linear body segment.

Applying Eq. (6) to all four interpolation points in FIGURE 2 gives an expression $\{a\} = [A]^{-1}\{v\}$ for the unknown coefficients $a_i, i = 0,...,3$, where $[A]^{-1}$ is a coefficient matrix and $\{v\}$ is a vector with the normal velocity $\nabla\varphi \cdot \vec{n}(x_b, y_b)$ in the ghost node (x_g, y_g) and in the three mirror points (x_{1-3}, y_{1-3}).

The normal velocity $\nabla\varphi(x_b, y_b) \cdot \vec{n}(x_b, y_b)$ at the immersed boundary can then be written as the vector product $\{1 \; s \; s^2 \; s^3\}[A]^{-1}\{v\} \equiv \{b\}\{v\}$, where $\{b\}$ is a vector with four coefficients $b_i, i = 0,...,3.$ Using Eq. (4) to express the components of $\{v\}$, and applying the body-boundary condition in Eq. (5), the Neumann condition for the ghost node can be written as

$$\begin{aligned}
&\sum_{i=1}^{8}\left[\sum_{j=1}^{8} c_{j,i}^{k0}\nabla f_j\left(x_g^{k0}, y_g^{k0}\right)\cdot\vec{n}\left(x_b, y_b\right)\right]\varphi_i^{k0} + \\
&\frac{b_1}{b_0}\sum_{i=1}^{8}\left[\sum_{j=1}^{8} c_{j,i}^{k1}\nabla f_j\left(x_1^{k1}, y_1^{k1}\right)\cdot\vec{n}\left(x_b, y_b\right)\right]\varphi_i^{k1} + \\
&\frac{b_2}{b_0}\sum_{i=1}^{8}\left[\sum_{j=1}^{8} c_{j,i}^{k2}\nabla f_j\left(x_2^{k2}, y_2^{k2}\right)\cdot\vec{n}\left(x_b, y_b\right)\right]\varphi_i^{k2} + \\
&\frac{b_3}{b_0}\sum_{i=1}^{8}\left[\sum_{j=1}^{8} c_{j,i}^{k3}\nabla f_j\left(x_3^{k3}, y_3^{k3}\right)\cdot\vec{n}\left(x_b, y_b\right)\right]\varphi_i^{k3} = \\
&\frac{1}{b_0}\nabla\varphi\left(x_b, y_b\right)\cdot\vec{n}\left(x_b, y_b\right) = \frac{\vec{U}\cdot\vec{n}\left(x_b, y_b\right)}{b_0}
\end{aligned} \quad . \tag{7}$$

The indices $k0-k3$ denote the cells associated with the ghost node and interpolation points 1 – 3, see FIGURE 2: $k0$ is the cell that contains the ghost node and is closest to the fluid domain, whereas $k1-k3$ are the fluid cells where the interpolation points 1 – 3 are closest to the cell center.

Eq. (7) is enforced for all ghost nodes (layer 1 – 3) inside the body at every time step. However, if a ghost node is very close to the body boundary, the interpolation scheme can cause numerical inaccuracies due to very densely spaced interpolation points. So, when the normal distance from a ghost node to the body boundary s is smaller than some predefined tolerance s_{tol}, Eq. (7) is replaced with the lowest-order approximation possible, i.e. the body-boundary condition is enforced directly on the ghost node,

$$\sum_{i=1}^{8}\left[\sum_{j=1}^{8} c_{j,i}^{k0}\nabla f_j\left(x_g^{k0}, y_g^{k0}\right)\cdot\vec{n}\left(x_b, y_b\right)\right]\varphi_i^{k0} = \vec{U}\cdot\vec{n}\left(x_b, y_b\right). \tag{8}$$

$s_{tol} = 0.02\sqrt{\Delta x^2 + \Delta y^2}$ is found to be a reasonable choice, where Δx and Δy are the grid-separation distances in x and y direction, respectively. Eq. (8) is expected to have small influence on the overall accuracy, since $s > s_{tol}$ for most ghost nodes.

Assembly of Global Matrix System

Eq. (3), (4), (7) or (8) and eventual Dirichlet conditions constitute a global matrix equation for the velocity potential in all grid nodes. This has a narrow-banded coefficient matrix with dimension $(n_x n_y, n_x n_y)$, where n_x and n_y are the number of grid nodes in x and y direction, respectively. An example of the structure of a coefficient matrix in this system is shown in FIGURE 3. When the HPC method is applied with a body-fitted grid, there are at most nine nonzero elements in each row of the global coefficient matrix. These are indicated as “Basic HPC” in the figure. With the applied IBM, there can be up to 32 nonzero elements in the rows connected with ghost nodes inside the body. These elements do not deviate much from the diagonal matrix. Moreover, the number of ghost nodes will always be relatively small compared to the total number of nodes, so that the matrix is still essentially a diagonal matrix. Therefore, the IBM does not compromise the computational efficiency that is achievable with the HPC method due to its sparse coefficient matrix significantly.

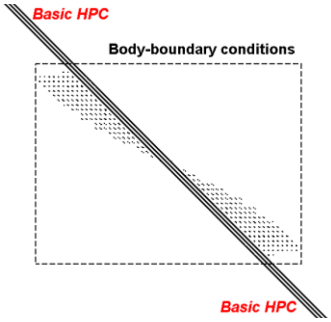


FIGURE 3: Nonzero entries in global coefficient matrix. Rows inside the stapled square correspond to the enforcement of the body-boundary condition at the ghost nodes.

An iterative solver applying the GMRES algorithm and with a tolerance 0.5E-08 is used to solve the global equation system at each time step in the present work.

Calculation of Fluid Pressure

The fluid pressure follows from Bernoulli's equation [13],

$$p(x,y,t) = -\rho\frac{\partial\varphi}{\partial t} - \frac{1}{2}\rho\left|\nabla\varphi\right|^2 , \tag{9}$$

where the hydrostatic pressure term is omitted. The quadratic (last) term in Eq. (9) is easily computed for any point by taking the analytical spatial gradient of Eq. (1), whereas a backward Finite Difference Method (FDM) is used to evaluate the linear term. A second-order FDM uses the value of the velocity potential at the present and previous time step,

$$\frac{\partial\varphi(x,y,t)}{\partial t} \approx \frac{\varphi(x,y,t)-\varphi(x,y,t-\Delta t)}{\Delta t} , \tag{10}$$

where Δt is the time step. A third-order approximation can be obtained by fitting a Lagrange polynomial to the velocity potential, $\varphi \approx c_1 + c_2\Delta t + c_3(\Delta t)^2 + c_4(\Delta t)^3$, using the value of the velocity potential at the present and the previous three time steps to determine the unknown coefficients. The linear pressure is then given as

$$\frac{\partial\varphi(x,y,t)}{\partial t} \approx c_2 + 2c_3\Delta t + 3c_4\left(\Delta t\right)^2 , \tag{11}$$

which better approximates the temporal curvature of the linear pressure. However, as illustrated later it may cause additional pressure oscillations compared to Eq. (10) for moving bodies.

Integration of the Pressure

The fluid pressure at the body boundary is interpolated by using a similar scheme as for the normal velocity in Eq. (6), i.e.

$$p(r) = d_0 + d_1 r + d_2 r^2 + d_3 r^3 . \tag{12}$$

The same interpolation points (separated by a distance $2s$) as used for the body-boundary condition are employed to determine the coefficients $d_i, i = 0,...,3$. The interpolation variable r is defined in FIGURE 2 as the distance from the relevant ghost node in the direction of the local body normal vector. The pressure in each interpolation point is taken as a weighted sum of the pressure in the nine grid nodes of the cell that has the interpolation point closest to its center node.

The pressure associated with ghost nodes in layers 1 and 2 (see FIGURE 1) is interpolated at the position of the immersed boundary with Eq. (12) and integrated over the body surface. The procedure for doing this is as follows: Each of the ghost nodes is projected normally onto the piecewise linear body surface, defining an intersection point. Then, each ghost node i is associated with an area dA_i of the body that extends from the intersection point for this particular ghost node and halfway towards the adjacent intersection points for neighbor ghost nodes on both sides (see FIGURE 4). Thus, in practice, a new body discretization which depends more on the number of ghost nodes than on the number of Lagrangian markers is defined for integrating the fluid pressure.

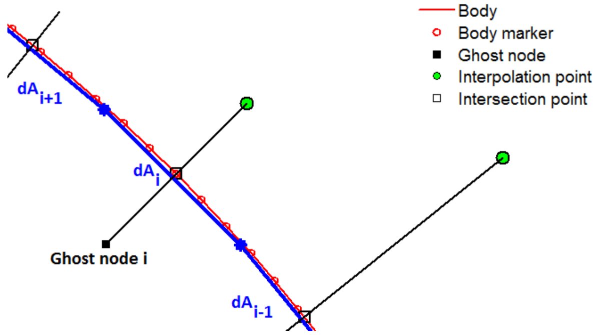


FIGURE 4: Assignment of pressure areas (blue curves) to ghost nodes.

Thus, with N ghost nodes in layer 1 and 2, the integrated pressure-force vector on the body is

$$\vec{F}(t) = -\sum_{i=1}^{N} p_i dA_i \vec{n}_i , \tag{13}$$

where p_i is the pressure at the immersed boundary interpolated from Eq. (12) and $\vec{n}_i$ is the normal vector directed out of the body. FIGURE 5 shows the convergence of the total integrated surface area and the variation of dA_i for a cylinder as a function of the number of grid nodes per cylinder diameter. The error in the total area decreases at a rate $(D/\Delta x)^{-2.2}$ as the grid density, and thus the number of ghost nodes, increases. At the same time, the dA_i values become more homogenous.

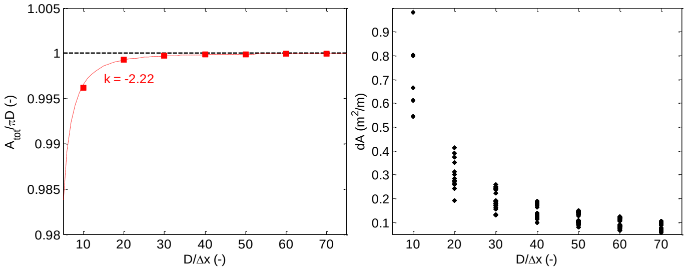


FIGURE 5: Left: Ratio between total area $A_{tot} = \sum dA_i$ and analytical area πD versus the number of grid nodes per cylinder diameter $D/\Delta x$. k is the exponent of the error. Right: Spread of incremental area versus $D/\Delta x$.

Numerical oscillations in the pressure force associated with sharp-interface IBMs are partly due to the role conversion of different grid nodes when the body moves over the grid [3]. In particular, when new grid nodes are activated, the time history of the velocity potential in these nodes does not have a physical meaning. This causes nonphysical values in the FDM scheme used to estimate the linear pressure term in Eq. (9). The deficiency is remedied in the present method by imposing the body-boundary condition in Eq. (7) on all three layers of ghost nodes, while only using the two outer layers in the pressure integration scheme. This is done because nodes that emerge from the interior region into the third ghost-node layer do not have a time history directly related to the exterior flow. Differently, when nodes emerge from the third layer into the second layer of ghost nodes, the velocity potential at the previous time step was physically consistent with the exterior flow through the imposed boundary condition. This puts certain restrictions on the grid density and the time step, i.e. the body is not allowed to move more than one grid spacing per time step if Eq. (10) is applied or more than one third of the grid spacing per time step if Eq. (11) is applied.

It should be noted that other authors have used different treatments to reduce spurious oscillations associated with numerical time derivatives. Liu & Hu [12] used an extrapolation scheme based on pseudo-points inside the body to reconstruct the fluid variables. Lee et al. [3] added artificial source terms in the momentum equation, thereby reducing force oscillations on a moving body by better preserving global conservation laws. Seo & Mittal [4] also identified mass sources associated with the generation of fresh and dead cells as the main cause of spurious oscillations, and applied a cut-cell based approach to mitigate them.

NUMERICAL RESULTS FOR A TWO-DIMENSIONAL CYLINDER IN INFINITE FLUID

Fixed Cylinder in Oscillatory Flow

We first consider a fixed cylinder with radius $R = 5m$ in an oscillatory flow, where the ambient flow is given as

$$\nabla\varphi(x,y,t) = \vec{i}U_a\cos\omega t \qquad (14)$$

with $U_a = 1ms^{-1}$ and $\omega = 0.5s^{-1}$. The body-boundary condition on the cylinder surface is given as $\partial\varphi/\partial n = G = 0$. A square computational domain is applied, with the cylinder located in the center of the domain (see FIGURE 6). The higher-order FDM in Eq. (11) is applied to compute the linear pressure term $-\rho\partial\varphi/\partial t$. The velocity potential for this case has an analytical solution for all points in the fluid,

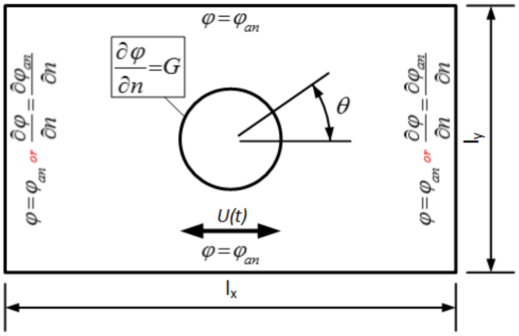
$$\varphi(r,\theta,t) = U_a\cos\omega t\left(r + \frac{R^2}{r}\right)\cos\theta\,. \qquad (15)$$

Eq. (15) is given in polar coordinates, with $r = \sqrt{x^2+y^2}$ and $\theta = \cos^{-1}(x/r)$. The corresponding analytical force on the cylinder is

$$f_x/\rho = -2\pi\omega R^2 U_a\sin\omega t\,. \qquad (16)$$

f_x is the force per unit span of the two dimensional cylinder with unit Nm^{-1}.

Two types of boundary conditions are considered along the outer boundaries of the computational domain in FIGURE 6. In the first case, the analytical solution of the velocity potential in Eq. (15) is given as Dirichlet condition along all four boundaries. In the second case, the analytical solution is given as Dirichlet condition along the upper and lower boundaries and as Neumann condition along the left and right boundaries. An uniform grid with grid spacing $0.25m \le \Delta x \le 1.0m$ is applied. The time step is constant, $\Delta t = 0.05s$, i.e. 251 time steps per oscillation period. Thus, the maximum Courant-Friedrichs–Lewy (CFL) number $U_a\Delta t/\Delta x$ varies between 0.05 - 0.25.



$\varphi=\varphi_{an}$
$\frac{\partial\varphi}{\partial n}=G$
θ
$\varphi=\varphi_{an}$ or $\frac{\partial\varphi}{\partial n}=\frac{\partial\varphi_{an}}{\partial n}$
$\varphi=\varphi_{an}$ or $\frac{\partial\varphi}{\partial n}=\frac{\partial\varphi_{an}}{\partial n}$
l_y
U(t)
$\varphi=\varphi_{an}$
l_x

FIGURE 6: Computational domain for two-dimensional cylinder in oscillatory flow.

As a measure of the accuracy of the numerical results, we compute the L2-error

$$L_{2,\alpha} = \sum_{i=1}^{K}\left(\alpha_{num,i} - \alpha_{an,i}\right)^2 \Big/ \sum_{i=1}^{K}\alpha_{an,i}{}^2 \qquad (17)$$

for the variable $\alpha = \varphi$ and $\alpha = f_x$. More in detail, $L_{2,\varphi}$ is computed for a given time step with $K = N$ the number of grid nodes (disregarding the interior nodes, see FIGURE 1), whereas L_{2,f_x} is computed over the entire time series with $K = M$ the number of time steps.

The L2-errors are plotted in FIGURE 7 versus the number of grid nodes per cylinder diameter $D/\Delta x$. The number of Lagrangian body-marker points n_b is varied, so that the ratio of the uniform distance between marker points $ds_b = \pi D/n_b$ to the grid spacing Δx is constant, $ds_b/\Delta x \approx 10$. The convergence of the velocity potential is seen to be between third- and fourth-order in the grid spacing, $L_{2,\varphi} \sim (D/\Delta x)^{-3.7}$. This agrees well the theoretical order of the applied HPC method in Δx [6], [7], and with the body-boundary condition interpolation scheme in Eq. (6) which is third-order in the normal coordinate r. From FIGURE 1, for most of the ghost nodes the distance between the associated interpolation points ($2s$, see FIGURE 2) is so that $r = O(\Delta x)$. The error in the integrated force converges at a rate $L_{2,f_x} \sim (D/\Delta x)^{-2.3}$, which is slower than for the error in the velocity potential. This seems to be connected with the estimation of the body surface area, which converge at a similar rate towards its analytical value in FIGURE 5. This is encouraging in the sense that refining the area-integration scheme likely can improve the convergence of the force.

The results in FIGURE 7 indicate that the proposed IBM does not compromise the overall convergence of the fundamental quantity (the velocity potential) in the HPC method. The integrated force has better than second-order convergence, and if only the three data points with coarsest grids in FIGURE 7 are included, the convergence is of third order. For the highest grid densities, the convergence rates decrease somewhat. The reason behind this is not fully understood; it could possibly relate to small separation distances between ghost nodes and the immersed boundary when the grid spacing becomes very small.

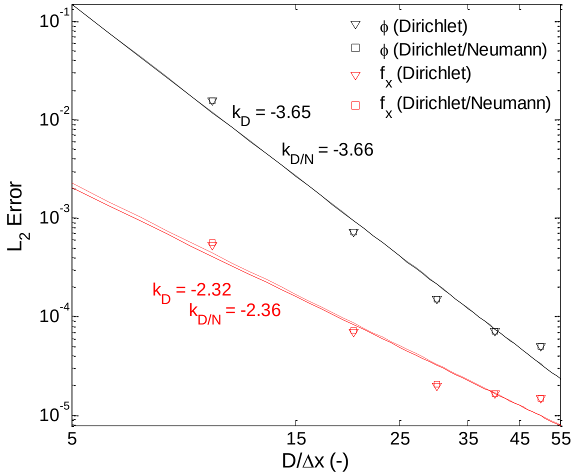


FIGURE 7: Convergence of velocity potential and longitudinal force on fixed cylinder in oscillatory flow. D: Dirichlet condition along domain boundaries, D/N: Dirchlet/Neumann conditions along domain boundaries.

It is expected that the convergence of the velocity potential shown in FIGURE 7 also applies for other types of geometries where the body-curvature varies smoothly. In case of sharp corners, where the potential-flow solution becomes singular, additional measures such as the local analytical solution proposed by Liang et al. [10] would be necessary to retain an accurate solution.

Oscillating Cylinder in Undisturbed Fluid

The main motivation behind the proposed IBM is to remove the difficulties involved in creating and regenerating body-fitted grids in simulations with arbitrarily shaped moving bodies. It is therefore of interest to consider a harmonically oscillating cylinder in still fluid, and compare the performance of the IBM to the physically equivalent case of a fixed cylinder in oscillatory flow. Forced oscillations are considered, where the velocity is given by

$$\frac{dx(t)}{dt} = U_a \cos \omega t \qquad (18)$$

and the displacement estimated by numerical time integration. The analytical solution of the velocity potential in this case is

$$\varphi(r,\theta,t) = -U_a \cos \omega t \left(\frac{R^2}{r}\right) \cos\theta \,. \qquad (19)$$

The hydrodynamic force corresponds to a pure radiation load,

$$f_x/\rho = \pi\omega R^2 U_a \sin \omega t \,. \qquad (20)$$

First, we consider the convergence of f_x and compare it to the convergence for a fixed cylinder in oscillatory flow. The parameters are the same as in FIGURE 7, i.e.: $R = 5m$, $U_a = 1ms^{-1}$, $\omega = 0.5s^{-1}$and $\Delta t = 0.05s$. This gives a displacement amplitude $U_a/\omega = 2.0m = 0.2D$. We also compare with results for three other velocity amplitudes: $U_a = 0.5ms^{-1}$, $1.25ms^{-1}$ and $2.5ms^{-1}$. For the latter case, the time step is reduced to $\Delta t = 0.025s$ due to CFL requirements given by the higher-order FDM for the linear pressure in Eq. (11) ($CFL \le 0.33$). All other parameters are the same for each case. The grid is varied so that $0.25m \le \Delta x \le 1.0m$ in a square-computational domain with dimension $4D \times 4D$ and with Eq. (19) defined as Dirichlet condition along the outer boundaries. The ratio of the distance between body marker points to the grid spacing is $ds_b/\Delta x \approx 10$ for all grids.

FIGURE 8 shows that f_x overall converges slower for the oscillating cylinder compared to the fixed cylinder in oscillatory flow, and also that the magnitude of the error is larger for the former. Furthermore, the convergence rate decreases slightly while the magnitude of the error increases slightly for the largest oscillation amplitude.

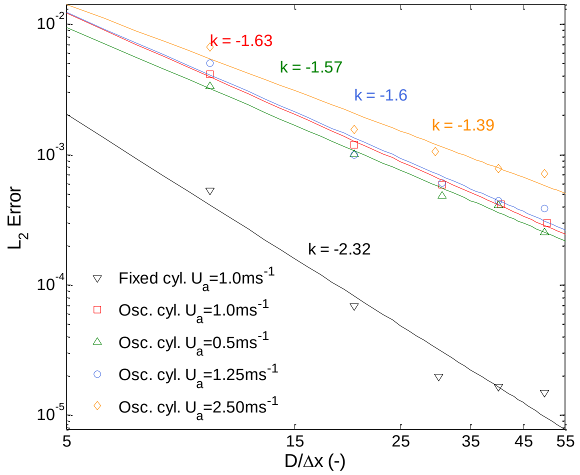


FIGURE 8: Convergence of force f_x on a fixed cylinder in oscillatory flow and on an oscillating cylinder in still fluid with four different oscillation amplitudes.

It is likely that the poorer accuracy and convergence for the oscillating cylinder compared to the fixed cylinder is due to spurious pressure oscillations, which lead to spurious oscillations also in the integrated force. An indication of this is seen by comparing the force on the oscillating cylinder with the force on a fixed cylinder multiplied by a factor -0.5 in FIGURE 9. The left column shows results for the oscillating cylinder with displacement amplitude $U_a/\omega = 0.2D$ for three different grid refinements, while the right column shows results for a displacement amplitude $U_a/\omega = 0.5D$. Severe oscillations can be seen for the coarsest grid, especially for the larger U_a/ω-value, making clear that a finer grid is needed. For the left column, the spurious oscillations converge quickly towards zero, and $\Delta x = 0.5m$ (middle figure) seems to be a sufficiently refined grid for practical purposes.

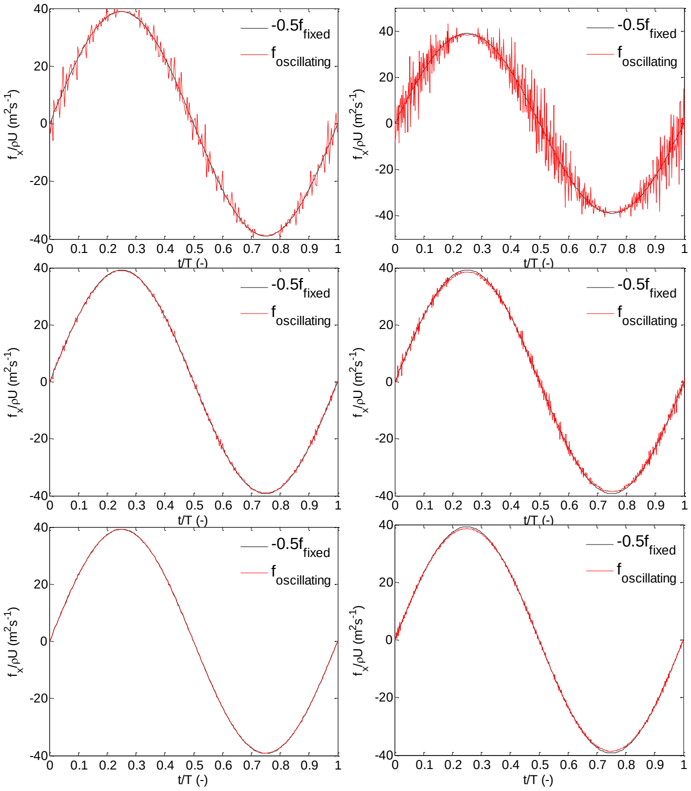


FIGURE 9: Time series of the force on a fixed cylinder in oscillatory flow and on a cylinder oscillating in still fluid. Top/mid/bottom: Δx=1.0/0.5/0.25m, left/right: U_a=1.0/2.5ms^{-1}. The time axis is normalized against the oscillation period $T = 2\pi/\omega$.

The observations from FIGURE 9 are consistent with the results by Lee et al. [3], where spurious force oscillations on an oscillating cylinder decreased with increasing CFL number. Although their study considered viscous flow, their conclusions are relevant also for potential-flow IBMs. They further found that the effect of refining the grid was stronger than that of increasing the time step. Seo & Mittal [4] made a similar observation; in their case the spurious force oscillations on an oscillating cylinder decreased both by decreasing the grid spacing and by increasing the time step. Again, the effect of the former was stronger.

Steadily Advancing Cylinder in Undisturbed Fluid

As illustrated for the oscillating cylinder above, IBMs can cause spurious pressure oscillations for moving bodies if the grid is not sufficiently dense (or if the time step is too fine compared to the grid density, effectively giving a very small CFL number). A body moving with a constant velocity is considered as the most challenging case in this respect. We therefore proceed to study a cylinder with $R = 5m$ that advances with a constant velocity $dx/dt = 0.5ms^{-1}$ in a $10D \times 6D$ computational domain. The fluid is otherwise at rest, with zero Neumann conditions along the vertical boundaries and zero Dirichlet conditions along the horizontal boundaries (recall FIGURE 6). The initial and final positions of the cylinder center are $x = -10m$ (i.e. $4D$ right of the left boundary) and $x = 10m$, respectively. We consider a situation with fixed time step, $\Delta t = 0.25s$, and the grid spacing is varied so that $0.4m \le \Delta x \le 1.0m$. This corresponds to CFL numbers ranging from 0.13 – 0.31. The ratio of the distance between Lagrangian body markers to grid spacing is kept constant, $ds_b/\Delta x \approx 10$.

Time series of the integrated force on the cylinder are shown in FIGURE 10 with the linear pressure evaluated by the low-order FDM in Eq. (10), and similarly in FIGURE 11 with the linear pressure term evaluated by the higher-order FDM in Eq. (11). The figures show that the forces oscillate around the analytical value (zero), with decreasing amplitudes for increasing grid refinement (i.e. increasing CFL number). The amplitudes approximately double in magnitude when the higher-order FDM for the linear pressure is used instead of the low-order FDM. Both the peak-to-peak amplitude and the standard deviation of the spurious oscillations in the longitudinal force have close to third-order convergence in Δx. Spurious oscillations in the transverse force are all insignificant but for the coarsest grid, and do not seem to follow any specific convergence pattern with respect to the grid density.

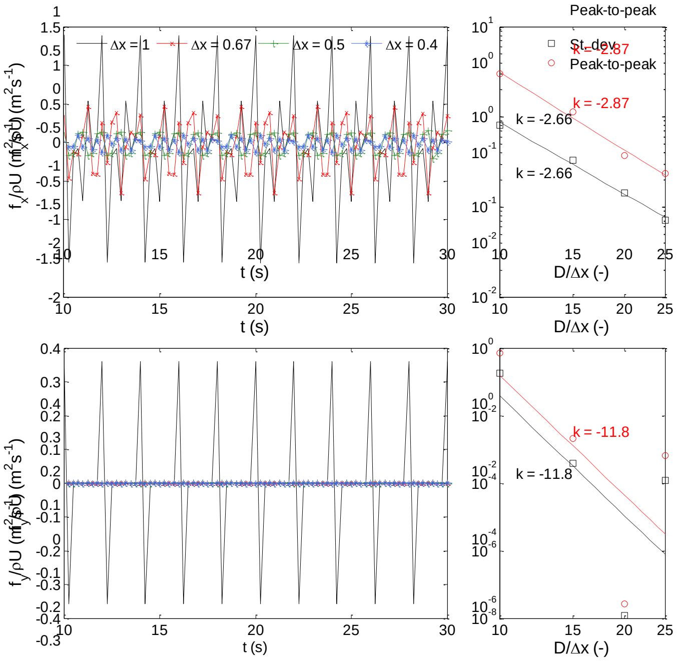


FIGURE 10: Left: Time series of forces on steadily advancing cylinder with fixed Δt and varying Δx. Right: Convergence plots for standard deviation and peak-to-peak amplitude of forces. Linear pressure evaluated with Eq. (10).

The spurious behavior in the force is related to oscillations in the fluid pressure. It is therefore instructive to examine the time series of the pressure in a point fixed in space as the cylinder advances. For this, we consider the point shown in

FIGURE 12, which is 1 m above the cylinder surface midway through the simulation. This is a fixed-grid node for all grid densities. The low-order FDM in Eq. (10) is used to evaluate the linear pressure. The pressure time series for the different Δx values are plotted in FIGURE 13. Both the total pressure and the separated components of Eq. (9) are shown. It is clear that the spurious pressure oscillations mainly originate from the linear pressure component, whereas the quadratic component is insensitive to grid refinements.

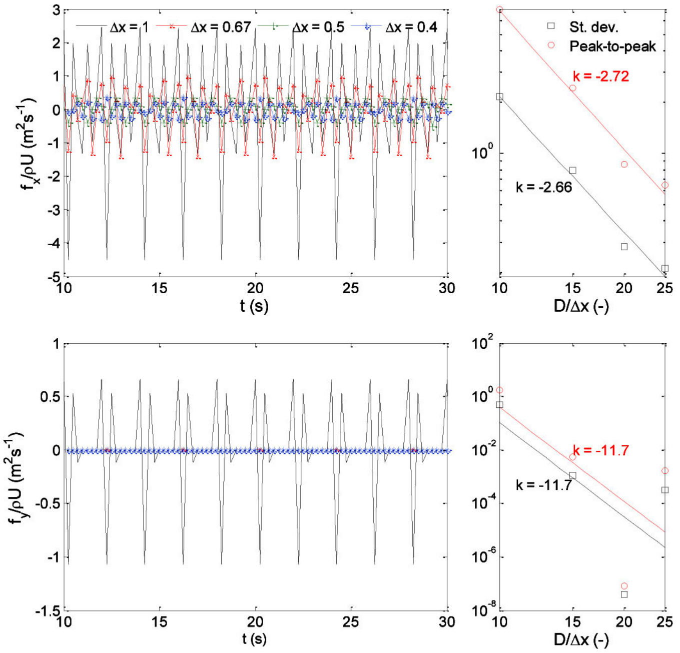


FIGURE 11: Same as FIGURE 10, but with linear pressure evaluated with Eq. (11).

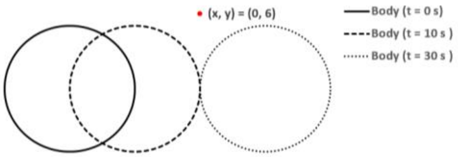


FIGURE 12: Fixed point for evaluation of pressure. The point is 6 m above the center of the cylinder at *t = 20s*.

Some remarks can be made concerning the practical significance of the observed oscillations. First, by carefully selecting an appropriate grid refinement and time step for the boundary velocities to be expected, the magnitude of the numerical oscillations can be reduced to a small portion of the physical quantity. A CFL number in the range 0.15 – 0.30 seems to be a good guidance in this respect. Secondly, the frequency of the spurious oscillations are in the order of Δt^{-1}, which in most cases will be significantly higher than any important frequency. Thus, in case of a freely moving rigid body with a large inertia, e.g. an offshore floater, this will act naturally as a filter, removing the spurious oscillations from the reaction time series. For a flexible body, however, higher frequencies may be present. In this case, special care may be necessary.

Numerically speaking, it is of course equally important to prevent growing instabilities in time-marching schemes due to small spurious oscillations. A possible mean to remove the problem of numerical time derivatives entirely in a potential flow is to solve an additional boundary-value problem similar to that illustrated in FIGURE 6 for the temporal derivative of the velocity potential. Since φ satisfies the Laplace equation initially, its time derivative $\partial\varphi/\partial t$ (the acceleration potential) will also do the same, and with the same global coefficient matrix as for the velocity potential. An additional matrix equation for $\partial\varphi/\partial t$ then has to be solved for each time step with the appropriate boundary conditions. For instance Greco [14] successfully used this idea to determine the acceleration potential in her BEM approach.

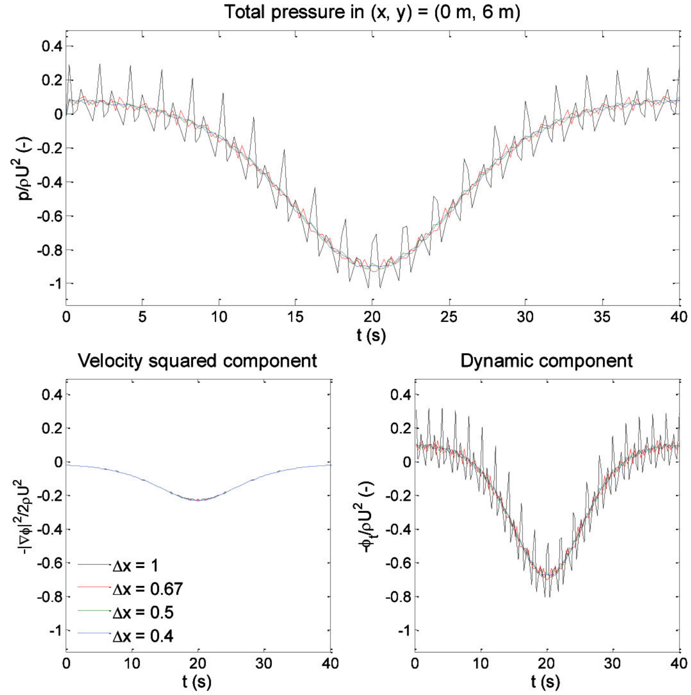


FIGURE 13: Time series of the pressure- and pressure components in the point shown in FIGURE 12 for four different CFL numbers.

DISCUSSION AND FURTHER WORK

The novel feature proposed in this paper is an immersed boundary method for potential-flow simulations by the HPC method with application to a moving body immersed in a fixed (Cartesian) background grid. The interpolation of fluid properties, that is necessary in an IBM, is natural in the HPC method since the velocity potential is developed as a combination of harmonic polynomials. A concept is introduced to ensure that fresh nodes do not introduce inaccuracies, which is important to prevent unphysical behavior when $-\rho\partial\varphi/\partial t$ is interpolated at the position of the immersed boundary.

No assumptions are made regarding the motion of the body, which means that motions of arbitrary amplitude can be simulated as long as inherent restrictions due to the CFL number and grid convergence are met.

A fixed cylinder in oscillatory flow was first studied, demonstrating that the proposed IBM maintains the spatial convergence properties of the velocity potential in the HPC method. Also the computational efficiency is substantially maintained. The force on the cylinder converges slower than the velocity potential, most likely due to the scheme applied to estimate the body-surface area in the pressure integration.

The force on an oscillating cylinder converged somewhat slower than that on a fixed cylinder in oscillatory flow. This is possibly due to oscillations in the fluid pressure associated with numerical time-differentiation schemes.

In general, it is found that the present method is able to compute forces on moving bodies accurately and with little contributions from spurious oscillations, as long as a reasonably fine background grid is applied. The required grid density is generally a function of the body curvature, the body velocity and the time step. Examined cases indicate that the required grid spacing is approximately proportional to the body velocity.

Very short time steps Δt generally tend to increase pressure oscillations associated with numerical time differentiation schemes, which is not strictly desirable from a convergence perspective. However, this is a recurring tendency also observed by other authors working with IBMs, and can be mitigated by adjusting the grid density accordingly. The CFL number, which can be considered as a non-dimensional body velocity, is a governing parameter in this context. It is found that CFL numbers in the range 0.15 – 0.30 represent a good compromise between accuracy and minimization of spurious oscillations.

Since the presented method does not perform any grid modifications, it should be straightforward to use also with other boundary-tracking methods, as well as to extend it to three-dimensional applications and to problems with elastic bodies. The next step is to attempt to use the proposed IBM to simulate fully nonlinear wave-body interactions for a floating body in a HPC-based numerical wave tank. Following this approach, both the floating body and the free surface can be immersed in the same Cartesian background grid. This allows for efficient computation of nonlinear free-surface flows combined with arbitrary body motions.

ACKNOWLEDGMENTS

This work has been carried out at the Centre for Autonomous Marine Operations and Systems (AMOS). The Norwegian Research Council is acknowledged as the main sponsor of AMOS. This work was supported by the Research Council of Norway through the Centres of Excellence funding scheme, Project number 223254 - AMOS.

REFERENCES

[1] Peskin C.S., Flow Patterns Around Heart Valves: A Numerical Method. Journal of Computational Physics, 10, pp.252-271, 1972.

[2] Udaykumar H.S., Mittal R., Rampunggoon P., Khanna A., A Sharp Interface Cartesian Grid Method for Simulating Flows with Complex Moving Boundaries. Journal of Computational Physics, 174, pp.345-380, 2001.

[3] Lee J., Kim J., Choi H., Yang, K-S., Sources of spurious force oscillations from an immersed boundary method for moving-body problems. Journal of Computational Physics, 230, pp.2677-2695, 2011.

[4] Seo J.H., Mittal R., A sharp-interface immersed boundary method with improved mass conservation and reduced spurious pressure oscillations. Journal of Computational Physics, 230, pp.7347-7363, 2011.

[5] Berthelsen P.A., Faltinsen O.M., A local directional ghost cell approach for incompressible viscous flow problems with irregular boundaries. Journal of Computational Physics, 227, pp.4354-4397, 2008.

[6] Shao Y.L., Faltinsen O.M., Towards efficient fully-nonlinear potential-flow solvers in marine hydrodynamics. Proceedings of the 31st International Conference on Ocean, Offshore and Arctic Engineering (OMAE). Rio de Janeiro, Brazil, 2012.

[7] Shao Y.L., Faltinsen O.M., A Harmonic Polynomial Cell (HPC) Method for 3D Laplace Equation with Application in Marine Hydrodynamics. Journal of Computational Physics, 274, pp.312-332, 2014.

[8] Shao Y.L., Faltinsen O.M., Fully-nonlinear wave-current-body interaction analysis by a harmonic polynomial cell (HPC) method. Journal of Offshore Mechanics and Arctic Engineering, 136, 2014.

[9] Fredriksen A., Kristiansen T., Faltinsen O.M., Experimental and numerical investigation of wave resonance in moonpools at low forward speed. Applied Ocean Research, 47, pp.28-46, 2014.

[10] Liang H., Faltinsen O.M., Shao Y.L., Application of a 2D harmonic polynomial cell (HPC) method to sloshing, singular flows and lifting problems. Under review in Applied Ocean Research.

[11] Tseng Y-H., Ferziger J.H., A ghost-cell immersed boundary method for flow in complex geometry. Journal of Computational Physics, 192, pp.593-623, 2003.

[12] Liu C., Hu C., An efficient immersed boundary treatment for complex moving object. Journal of Computational Physics, 274, pp.654-680, 2014.

[13] Faltinsen O.M., Sea Loads on Ships and Offshore Structures. Cambridge University Press, 1990.

[14] Greco M., A two-dimensional Study of Green-Water Loading. PhD Thesis, Norwegian University of Science & Technology (NTNU), 2001.